tn

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/944,776	08/31/2001	Andrej Kocev	200301967-2	1813

22879 7590 03/24/2006

HEWLETT PACKARD COMPANY
P O BOX 272400, 3404 E. HARMONY ROAD
INTELLECTUAL PROPERTY ADMINISTRATION
FORT COLLINS, CO 80527-2400

EXAMINER	
PHAM, THOMAS K	
ART UNIT	**PAPER NUMBER**
2121	

DATE MAILED: 03/24/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

MAILED

MAR 24 2006

Technology Center 2100

Application Number: 09/944,776
Filing Date: August 31, 2001
Appellant(s): KOCEV ET AL.

Michael R. Reinemann
For Appellants

EXAMINER'S ANSWER

This is in response to the appeal brief filed 02/09/2006 appealing from the Office action mailed 09/13/2005.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

The amendment after final rejection filed on 02/09/2006 has been entered.

Claims 37-39 are now allowed, therefore, will not be part of the claims on appeal.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6,718,413	Wilson et al.	4-2004
6,119,185	Westerinen et al.	9-2000
6,219,734	Wallach et al.	4-2001
6,085,276	VanDoren et al.	7-2000

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

1. Claims 13-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,119,185 ("Westerinen") in view of U.S. Patent No. 6,718,413 ("Wilson") and further in view of U.S. Patent no. 6,219,734 ("Wallach").

Regarding claim 13

Westerinen teaches a method for programmably allocating system resources to accommodate I/O transactions at I/O ports (see abstract) of a multiprocessor computer system (see col. 1 line 61) comprising:

- setting criteria (configuring) for routing transactions (tasks) to the port with respect to the numbers of devices at the ports (see col. 2 lines 43-53 and col. 7 lines 21-33); and
- with respect to the numbers of devices at the ports, assigning devices to the ports (see col. 8 lines 25-37).

Westerinen does not teach determining the number of devices being serviced via the ports; identifying at least one assemblies for hot swapping; and copying the contents of cache memories associated with the at least one identified assemblies.

However, Wilson teaches determining the number of devices being serviced via the ports (col. 10 lines 34-42) for the purpose prioritizing the devices to reduce the number of interrupts (see col. 10 lines 29-30).

Furthermore, Wallach teaches identifying at least one assemblies for hot swapping (col. 17 lines 33-35); and copy the contents of old adapter to the new added adapter [port and adapter since they are both providing communication to I/O devices] (col. 12 lines 60-64) for the purpose of keeping track and allocates resources for every managed adapter (see col. 9 lines 18-20).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the method of determining number of devices of Wilson with the method of Westerinen because it would provide for the purpose prioritizing the devices to reduce the number of interrupts.

Furthermore, it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the hot swapping method of Wallach with the method of Westerinen because it would provide for the purpose of keeping track and allocates resources for every managed adapter [port].

Regarding claim 15

Westerinen teaches a system for programmably allocating system resources to accommodate I/O transactions at I/O ports (see abstract) of a multiprocessor computer system (see col. 1 line 61), the system comprising:

- means for setting criteria (configuring) for routing transactions (tasks) to the port with respect to the numbers of devices at the ports (see col. 2 lines 43-53 and col. 7 lines 21-33); and
- means, responsive to the criteria, for assigning devices to the ports (see col. 8 lines 25-37).

Westerinen does not teach the means for determining the number of devices being serviced via the ports; at least one assemblies identified for hot swapping; and means for copying the contents of cache memories associated with the at least one identified assemblies.

However, Wilson teaches determining the number of devices being serviced via the ports (col. 10 lines 34-42) for the purpose prioritizing the devices to reduce the number of interrupts (see col. 10 lines 29-30).

Furthermore, Wallach teaches at least one assemblies identified for hot swapping (col. 17 lines 33-35); and copy the contents of old adapter to the new added adapter [port and adapter since they are both providing communication to I/O devices] (col. 12 lines 60-64) for the purpose of keeping track and allocates resources for every managed adapter (see col. 9 lines 18-20).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the system for determining number of devices of Wilson with the system of Westerinen because it would provide for the purpose prioritizing the devices to reduce the number of interrupts.

Furthermore, it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the hot swapping system of Wallach with the system of Westerinen because it would provide for the purpose of keeping track and allocates resources for every managed adapter [port].

Regarding claims 14 and 16

Wallach teaches assigning resources to the ports comprises at least one of assigning control registers to the ports, assigning direct memory access engines to the ports, assigning cache memory to the ports and assigning priorities among the transactions at the ports (col. 10 lines 58-61, "The configuration manager 500 … adapter's configuration space registers").

Regarding claims 17 and 19

Westerinen teaches a system determining the number and types of transactions anticipated at the ports, wherein the assignment of resources is further with respect to the numbers and types of transactions at the ports (col. 5 lines 38-46, "The Configuration Ruleset is … provides a "parallel" implementation").

Regarding claims 18 and 20

Wallach teaches the at least one identified assembly has a memory system, and the method further comprises copying the states and status of the memory systems associated with at least one identified assembly (col. 9 lines 22-25, "The registers of an adapter 310 … the status of the adapter").

2. Claims 21-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,119,185 ("Westerinen") in view of U.S. Patent No. 6,718,413 ("Wilson").

Regarding claim 21

Westerinen teaches a method for programmably allocating resources for processing Input/Output (I/O) transactions at a plurality of I/O ports of an I/O bridge (see abstract), the method comprising:

- setting criteria (configuring) for the transactions (tasks) at the at least one port with respect to the numbers of I/O devices being service at the ports (see col. 2 lines 43-53 and col. 7 lines 21-33); and
- assigning the resources to the at least one I/O port in response to the criteria (see col. 8 lines 25-37).

Westerinen does not teach identifying the number of I/O devices being serviced by at least one I/O ports.

However, Wilson teaches identifying the number of I/O devices being serviced by at least one I/O ports (col. 10 lines 34-42) for the purpose prioritizing the devices to reduce the number of interrupts (see col. 10 lines 29-30).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the method of determining number of devices of Wilson with the method of Westerinen because it would provide for the purpose prioritizing the devices to reduce the number of interrupts.

Regarding claim 22

Westerinen teaches assigning a plurality of direct memory access (DMA) engines for use in processing I/O transactions (see col. 11 lines 7-11)

Regarding claim 23

Westerinen teaches apportioning a selected number of DMA engines to process a given transaction at a particular I/O port (see col. 5 lines 38-42).

Regarding claim 24

Westerinen teaches apportioning at least one DMA engine to process at least one transaction at a port (see col. 7 lines 21-33).

Regarding claim 25

Westerinen teaches apportioning one DMA engine to process a given transaction at a port identified as servicing multiple I/O devices (see col. 5 lines 38-42).

3. Claims 26-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Westerinen in view of Wilson and further in view of U.S. Patent no. 6,085,294 ("VanDoren-94").

Regarding claim 26

Westerinen and Wilson teach a system for allocating resources but do not teach assigning at least one miss address file (MAF) value for processing I/O transactions. However, VanDoren-94 teaches at least one miss address file (MAF) (fig. 2 element 86a). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the MAF of VanDoren-94 with the system of

Westerinen and Wilson because it would provide for processing I/O transactions data which has not yet completed by the CPU.

Regarding claim 27

VanDoren-94 teaches assigning a plurality of miss address file (MAF) values for processing I/O transactions (col. 7 lines 38-39, "Each CPU 12a-12d … (MAF) 86a-86d").

Regarding claim 28

Westerinen, Wilson and VanDoren-94 teach a system for allocating resources with at least one MAF but do not teach reducing the assigned number of MAF. However, it would have been obvious to one of ordinary skill in the art at the time the invention was made to reduce or increase the number of MAF in accordance with the number of CPUs used.

4. Claims 29-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Westerinen in view of Wilson and further in view of U.S. Patent no. 6,085,276 ("VanDoren-76").

Regarding claim 29

Westerinen and Wilson teach a system for allocating resources with the I/O bridge but do not teach configuring to utilize a plurality of virtual channels to communicate with at least one processors of a multiprocessor computer system, and the resources include flow control credits associated with each of the plurality of virtual channels. However, VanDoren-76 teaches a plurality of virtual channels to communicate with the multiprocessor system, and the resources include flow control credits associated with

each of the plurality of virtual channels (col. 14 line 66 to col. 15 line 5, " Virtual channels are a scheme … among messages in the system"). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the virtual channels of VanDoren-76 with the system of Westerinen and Wilson because it would provide for eliminating flow-dependence and resource dependence cycles among messages in the system in order to eliminating deadlock in the cache coherence protocol.

Regarding claim 30

VanDoren-76 teaches setting the number of flow control credits associated with each virtual channel (col. 20 lines 14-20, "flow control from the … in the SMP system").

5. Claim 31 is rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,119,185 ("Westerinen") in view of U.S. Patent No. 6,718,413 ("Wilson") and further in view of U.S. Patent No. 6,219,734 ("Wallach").

Regarding claim 31

Westerinen and Wilson do not teach the I/O bridge comprises at least one control register, the at least one control register having a plurality of fields, and at least one field of the control register being associated with a corresponding resource, and the method further comprises writing to a selected field of the at least one control register so as to modify the assignment of resources.

However, Wallach teaches the I/O bridge comprises at least one control register, the at least one control register having a plurality of fields, and at least one field of the

control register being associated with a corresponding resource, and the method further comprises writing to a selected field of the at least one control register so as to modify the assignment of resources (col. 10 lines 57-61, "Once an adapter 310 … configuration space registers") for the purpose of keeping track and allocates resources for every managed adapter (see col. 9 lines 18-20).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the hot swapping method of Wallach with the method of Westerinen and Wilson because it would provide for the purpose of keeping track and allocates resources for every managed adapter [port].

6. Claims 32, 33, 40 and 41 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,219,734 ("Wallach") in view of U.S. Patent No. 6,119,185 ("Westerinen").

Regarding claim 32

Wallach teaches an Input/Output (I/O) bridge for use in a computer system having a plurality of processors, the I/O bridge comprising: a plurality of I/O ports, each I/O port configured to communicate with at least one I/O device that generates or receives transactions (col. 5 lines 1-5, "a programmable mass storage adapter … the operational computer"); resources for use in servicing the transactions of the I/O devices (col. 10 lines 57-61, "Once an adapter 310 … configuration space registers").

Wallace does not teaches a programmable logic configured and arranged to assign the resources among the I/O ports in response to the number of I/O devices with which the I/O ports are communicating.

However, Westerinen teaches a configuration logic that assign the resources among the I/O ports in response to the number of I/O devices (see col. 1 line 66 to col. 2 lines 12, "One embodiment of the .... Devices to that resource") for the purpose of efficiently and intelligently configures to achieve enhance performance and minimize conflicts (see col. 1 lines 49-52).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the configuration logic of Westerinen with the system of Wallach because it would provide for the purpose of efficiently and intelligently configures to achieve enhance performance and minimize conflicts.

Regarding claim 33

Westerinen teaches the resources comprise at least one direct memory access (DMA) engine configured to process the transactions (see col. 11 lines 7-11), and the programmable logic apportions the at least one of DMA engine to process at least one transaction at a given I/O port in response to the number of I/O devices coupled to the given I/O port (see col. 7 lines 21-33).

Regarding claim 40

Wallach teaches the configuration manager 500 comprises at least one control register associated with each I/O port, and the at least one control register has a first field for apportioning (col. 10 lines 58-61, "The configuration manager 500 ... adapter's

configuration space registers"). Westerinen teaches at least one apportioning at least one DMA engine to process at least one transaction at a port (see col. 7 lines 21-33).

Regarding claim 41

Wallace teaches the configuration manager 500 re-assigns resources among the I/O ports dynamically while the I/O bridge continues to operate (col. 10 lines 57-61, "Once an adapter 310 ... configuration space registers").

7. Claims 34-36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Wallach in view of Westerinen and further in view of U.S. Patent no. 6,085,276 ("VanDoren-76").

Regarding claim 34

Wallach and Westerinen teach a system for allocating resources but do not teach the resources include a plurality of miss address file (MAF) values for use in requesting information from the computer system, and the programmable logic sets the number of available MAF values. However, VanDoren-94 teaches assigning a plurality of miss address file (MAF) values for processing I/O transactions (col. 7 lines 38-39, "Each CPU 12a-12d ... (MAF) 86a-86d"). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the MAF of VanDoren-94 with the system of Wallach and Westerinen because it would provide for processing I/O transactions data which has not yet completed by the CPU. Furthermore, it is obvious to one of ordinary skill in the art at the time the invention was made to reduce or increase the number of MAF in accordance with the number of CPUs used.

Regarding claim 35

Wallach and Westerinen teach a system for allocating resources with the I/O bridge and a configuration manager 500 but do not teach the I/O bridge communicates with the computer system through a plurality of virtual channels, the resources include a plurality of flow control credits associated with the virtual channels, and the programmable logic assigns a number of flow control credits to each virtual channel. However, VanDoren-76 teaches a plurality of virtual channels to communicate with the multiprocessor system, and the resources include flow control credits associated with each of the plurality of virtual channels (col. 14 line 66 to col. 15 line 5, " Virtual channels are a scheme … among messages in the system") and setting the number of flow control credits associated with each virtual channel (col. 20 lines 14-20, "flow control from the … in the SMP system"). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the virtual channels of VanDoren-76 with the system of Wallach and Westerinen because it would provide for eliminating flow-dependence and resource dependence cycles among messages in the system in order to eliminating deadlock in the cache coherence protocol.

Regarding claim 36

VanDoren-76 teaches the virtual channels comprise a Request channel, a Read I/O channel, and a Write I/O channel (col. 15 lines 16-28, "a Q0 channel for carrying … from a processor to an IOP").

(10) Response to Argument

A. Appellants argue that cited reference failed to teach: "determining the number of devices being serviced by the ports, setting criteria for transactions at the port with respect to the number of devices, and with respect to the number of devices at the ports, assigning resources to the ports."

In response to appellants' arguments, the reference of Westerinen (USPN 6,119,185) discloses the purpose of the invention is to correctly configure a computer platform to support different types of I/O add-in adapters such as ISA, EISA and/or PCI. A set of rules (criteria) is generated to define the configuration process as described below in column 2 lines 41-53. It should be noted that the functionality of an I/O adapter is to include any number of fix I/O ports for communication with external device as well known in the art.

> "The apparatus and method of the present invention invoke-a rule-based reasoning scheme for efficiently and intelligently configuring a computer system. A goal is to correctly configure a computer platform supporting ISA, EISA and/or PCI add-in adapters, and tune the resource assignments for performance. Knowledge on how to configure a system is gathered from sources such as experts who perform computer configurations and providers of devices and operating systems that are used in computers. From this information, Rules are generated which define the configuration process. The Rules based configuration program is generated by functionally grouping the rules into executable tasks such as "configuration" or "analysis" rules. "

Westerinen, further discloses in column 8 lines 25-37 that the rules are also based on the number of devices currently plugged in with the system and prohibit these devices from using the same configuration setting to avoid any conflicts between devices so that transactions can be routed to the ports as described below:

> "With respect to Rules 5 and 6 regarding sharing resource values, during the assignment process, all potential setting values must be checked to verify that another device has not been allocated the same value. If the value is already being used and the resource can not be shared between the devices, then the current device is prohibited from using that value. The implementation of this checking mechanism is straight forward for those resources that use a single integer value, such as IRQ's and DMA's. If there is another device using a setting value and both resources cannot share

> values, then a conflict exists. Checking conflicts for resources that use a range of values (memory and I/O port) requires a determination as to whether there is overlap between the ranges."

According to the above two cited portions above, Westerinen is clearly setting up a computer platform for supporting different I/O adapter using a set of rules (or criteria) with respect to a number of devices currently active to check for any potential conflict occurring between the devices so that devices can be shared (assigned) and transactions can be processed at the adapters. Thus, Westerinen teaches the claimed limitations "setting criteria for routing transactions to the ports with respect to the numbers of devices at the ports".

Furthermore, Westerinen discloses an assignment of the devices is dependent upon all other devices currently active at the I/O ports as described in column 8 lines 38-51 below:

> "With respect to Rule 7 regarding dependent resource assignments, an example of this is as follows. The system board's serial port functionality may have choices for COM1 and COM2, where COM1 lists an IRQ resource with the setting option of 4 and an I/O Port with a range option of 3F8h-3FFh and COM2 lists an IRQ resource with the setting option of 4 and an I/O Port with a range option of 2F8h-2FFh. In order to select COM1, both IRQ4 and I/O Port 3F8h-3FFh must be made. The assignment of one resource is "dependent" upon the other. If IRQ4 cannot be assigned because of a conflict, than I/O Port 3F8h-3FFh can not be assigned. Thus, both the assignment of both of these resources should be made at the same time to insure a valid selection at the completion of the configuration process."

Thus, Westerinen teaches the claimed limitations "with respect to the numbers of devices at the ports, assigning devices to the ports."

In addition, the reference of Wilson (USPN 6,718,413) discloses a method to determine the number of devices are contending for a bus to re-select a host adapter as described in column 10 lines 29-46 below:

> "FIG. 6 shows a flowchart of an exemplary method for generating reduced number of interrupts in accordance with another embodiment of the present invention. In this method, one or more I/O commands are received for transferring data between a host computer and one or more I/O devices in operation 602. Then, at each arbitration phase after a command completion, the host adapter monitors SCSI bus contention, in operation 604, to determine the number of devices arbitrating for the bus to re-select the host adapter. Next in operation 606, it is determined whether more than one device are contending for the bus. If so, data is transferred to the host adapter by a selected device

> having the highest priority (i.e., highest SCSI ID number) in operation 608. Then in operation 610, the SCSI devices including the host adapter wait for the completion of data transfer by the selected device. The completion of data transfer will be indicated by a command completion message asserted over the bus."

By combining the number of devices being serviced at a host adapter of Wilson with the system of Westerinen, the system of Westerinen can beneficially apply the method of Wilson for the purpose of generating reduced number of interrupts upon completing one or more commands.

In summary, for at least the reasons described above the combination of Westerinen and Wilson teaches the appellants' claimed limitations "determining the number of devices being serviced by the ports, setting criteria for transactions at the port with respect to the number of devices, and with respect to the number of devices at the ports, assigning resources to the ports."

B. Appellants argue that cited reference failed to teach: "assigning one of control registers, direct memory access engines or cache memories to ports."

In response to appellants' arguments, the reference of Wallace (USPN 6,219,734) discloses a configuration manager is responsible for managing all or some of a PC's adapters. The configuration manager also assigning a register to an adapter during a hot swap operation as described in column 9 lines 13-26 below:

> FIG. 5 is a block diagram illustrating the system components of the NetWare Operating System and an embodiment of the software components of the invention. A configuration manager 500 is responsible for managing all or some of the adapters on the PC buses 234 and 236 (FIG. 2), or 250, 252, 254 and 256 (FIG. 3). The configuration manager 500 keeps track of the configuration information for every managed adapter located on the fault tolerant computer system 100. The configuration manager 500 also allocates resources for every managed adapter and initializes each managed adapter's registers during a hot swap operation. The registers of an adapter 310 are components or intermediate memories whose values issues a certain action in the adapter, or whose values indicate the status of the adapter.

Since the limitation is to assign at least one of "control registers, direct memory access engines or cache memories to ports", Wallace teaches the claimed invention by assigning the control registers to the I/O adapters.

C. Appellants argue that cited reference failed to teach: "the assignment of resources with respect to the numbers and types of transactions at the ports."

In response to appellants' arguments, Westerinen discloses selecting different type of resources such as the IRQ or DMA for assigning to the I/O adapter as described below in column 5 lines 38-45:

> "The Configuration Ruleset is used to make resource assignments for a selected resource type (IRQ, DMA, etc.). The Agendas for the Configuration Ruleset are run in parallel using Agenda priority (discussed below) to determine the Rule's execution flow. For example, all active Rules in the "FindConflicts" Agenda are fired before any active Rules in the "Assign" Agenda. This provides a "parallel" implementation."

One of ordinary skill in the art would recognize that different types of transactions are determined depending upon the different types of resources in which the transactions are generated. Therefore, since the assignment of resources is based on the type of resources, the assignment of resources is also based on the type of transactions associated with the resources. Thus, the claimed limitation "the assignment of resources with respect to the numbers and types of transactions at the ports." is meet by the Westerinen reference.

D. Appellants argue that cited reference failed to teach: "programmable logic configured and arranged to assign the resources among the I/O ports in response to the number of I/O devices with which the I/O ports are communicating."

In response to appellants' arguments, the reference of Westerinen further teaches

Westerinen a program for assigning resources among the devices with the highest setting weight first. The program logic continues to loop through in response to all other requesting devices connecting among the I/O ports that are in communication with each other as described below in column 10 line 65 to column 11 line 11:

> A memory assigning cycle through steps 120-130 is generally as follows. A device requesting memory is pulled from the list (step 120). The legal settings for this device are investigated for a setting conflict (step 122). In step 124, the above ranking is undertaken and in step 126 a determination is made as to whether any of the ranked settings are available. In step 128, the device is assigned to the resource with the highest setting weight. In step 130, the program loops to consider the next memory requesting device or to consider the first I/O port requesting device. The program of steps 120-130 preferably loops until all I/O port requesting devices and then all DMA requesting devices (and any other devices) are accounted for. Step 132 indicates that the configuration is complete.

According to the above cited portion, Westerinen discloses a program is using a pre-defined logic (i.e. the resource with the highest setting weight) for assigning the resources in response to requesting devices among the I/O devices in communication. Thus, clearly, Westerinen teaches the claimed limitations "programmable logic configured and arranged to assign the resources among the I/O ports in response to the number of I/O devices with which the I/O ports are communicating."

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Thomas Pham

Anthony Knight
Supervisory Patent Examiner
Group 3600

Conferees:

Paul L. Rodriguez

Paul L. Rodriguez 3/20/06
Primary Examiner
Art Unit 2125

Anthony Knight